# Transliteration of Named Entity: Bengali and English as Case Study

**Asif Ekbal**[1] and **Sivaji Bandyopadhyay**[2]

[1,2]Department of Computer Science and Engineering, Jadavpur University, Kolkata-700032
Email: asif.ekbal @gmail.com [1] and sivaji_cse_ju@yahoo.com[2]

**Abstract**

This paper presents a modified joint-source channel model that is used to transliterate a Named Entity (NE) of the source language to the target language and vice-versa. As a case study, Bengali and English have been chosen as the possible source and target language pair. A number of alternatives to the modified joint-source channel model have been considered also. The Bengali NE is divided into Transliteration Units (TU) with patterns $C^{+}M$, where C represents a consonant or a vowel or a conjunct and M represents the vowel modifier or matra. An English NE is divided into TUs with patterns C*V*, where C represents a consonant and V represents a vowel. The system learns mappings automatically from the bilingual training sets of person and location names. Aligned transliteration units along with their contexts are automatically derived from these bilingual training sets to generate the collocational statistics. The system also considers the linguistic features in the form of possible conjuncts and diphthongs in Bengali and their corresponding representations in English. Experimental results of the 10-fold open tests demonstrated that the modified joint source-channel model performs best during Bengali to English transliteration with a Word Agreement Ratio (WAR) of 74.4% for person names, 72.6% for location names and a Transliteration Unit Agreement Ratio (TUAR) of 91.7% for person names, 89.3% for location names. The same model has demonstrated a WAR of 72.3% for person names, 70.5% for location names and a TUAR of 90.8% for person names, 87.6% for location names during back transliteration.

**Keywords:** Machine Transliteration, Named Entity, Named Entity Transliteration, Bengali Named Entity Transliteration, Joint-Source Channel Model and Modified Joint-Source Channel Model.

## Introduction

In Natural Language Processing (NLP) application areas such as information retrieval, question answering systems and machine translation, there is an increasing need to translate Out Of Vocabulary (OOV) words from one language to another. They are translated through transliteration, the method of translating into another language by expressing the original foreign word using characters of the destination language preserving the pronunciation in their original languages. Thus, the central problem in transliteration is predicting the pronunciation of the original word. Transliteration between two languages, that use the same set of alphabets, is trivial: the word is left as it is. However, for languages that use different characters, the names must be transliterated or rendered in the destination language alphabets. Technical terms and named entities make up the bulk of these OOV words. Named entities hold a very important place in NLP applications. Proper identification, classification and translation of named entities are very crucial in many NLP applications and pose a very big challenge to the NLP researchers. Named entities are usually not found in bilingual dictionaries and very generative in nature. Translation of named entities is tricky task: it involves both translation and transliteration. Transliteration is commonly used for named entities, even when the words could be translated.

The NE machine transliteration algorithms presented in this work have been evaluated with person names and location names. A machine transliteration system like this is very important in a multilingual country like India where large person name and location name collections like census data, electoral roll and railway reservation information must be available to multilingual citizens of the country in their own vernacular. The transliteration system presented here would be an effective tool for many NLP applications. Statistical machine translation systems can use such a system as a component to handle NE phrase translation in order to improve overall translation quality. Cross-Lingual Information Retrieval (CLIR) systems could identify relevant documents based on translation of NE phrases provided by such a system. Question Answering (QA) systems could benefit substantially from such a tool since the answer to many factoid questions involve NEs. In the present work, the various proposed models have been evaluated on the training sets of person names and location names.

A hybrid neural network and knowledge-based system to generate multiple English spellings for Arabic personal names is described in (Arbabi *et al.* 1994). (Knight & Graehl 1998) developed a phoneme-based statistical model using finite state transducer that implements transformation rules to do back-transliteration. (Stalls & Knight 1998) adapted this approach for back transliteration from Arabic to English for English names. A spelling-based model is described in (Al-Onaizan & Knight 2002a; Al-Onaizan & Knight 2002c) that directly maps English letter sequences

Copyright © 2007, American Association for Artificial Intelligence (www.aaai.org). All rights reserved.

into Arabic letters. The phonetics-based and spelling-based models have been linearly combined into a single transliteration model in (Al-Onaizan & Knight 2002b) for transliteration of Arabic named entities into English.

Several phoneme-based techniques have been proposed in the recent past for machine transliteration using transformation-based learning algorithm (Helen *et al.* 2001; Paola & Khudanpur 2003). (Nasreen & Larkey 2003) have presented a simple statistical technique to train an English-Arabic transliteration model from pairs of names. The two-stage training procedure first learns which n-gram segments should be added to unigram inventory for the source language, and then a second stage learns the translation model over this inventory. This technique requires no heuristic or linguistic knowledge of either language. (Goto *et al*. 2003) described an English-Japanese transliteration method in which an English word is divided into conversion units that are partial English character strings in an English word and each English conversion unit is converted into a partial Japanese Katakana character string. (Haizhou *et al*. 2004) presented a framework that allows direct orthographical mapping between English and Chinese through a joint source-channel model, called n-gram transliteration model.

A tuple n-gram transliteration model (Marino *et al*. 2005; Crego *et al*. 2005) has been log-linearly combined with feature functions to develop a statistical machine translation system for Spanish-to-English and English-to-Spanish translation tasks. The present work differs from (Goto *et al*. 2003; Haizhou *et al*. 2004) in the sense that identification of the transliteration units in the source language is done using regular expressions and no probabilistic model is used. Moreover the proposed model differs in the way the transliteration units and the contextual information are defined. No linguistic knowledge is used in (Goto *et al*. 2003; Haizhou *et al*. 2004) whereas the present work uses linguistic knowledge in the form of possible conjuncts and diphthongs in Bengali and their representations in English.

The proposed transliteration models are general and can be applied for any language pair. The transliteration unit (TU) alignment process is applicable for the languages that share a comparable orthography with English such as Bengali and other Indian languages.

## Machine Transliteration and Joint Source-Channel Model

A transliteration system takes as input a character string in the source language and generates a character string in the target language as output. The process can be conceptualized as two levels of decoding: segmentation of the source string into transliteration units and relating the source language transliteration units with units in the target language, by resolving different combinations of alignments and unit mappings. The problem of machine transliteration has been studied extensively in the paradigm of the noisy channel model.

For a given Bengali name B as the observed channel output, we have to find out the most likely English transliteration E that maximizes P (E|B). Applying Bayes' rule, it means to find E to maximize

$$P(B,E) = P(B|E) * P(E) \quad (1)$$

with equivalent effect. This is equivalent to modelling two probability distributions: P(B|E), the probability of transliterating E to B through a noisy channel, which is also called transformation rules, and P(E), the probability distribution of source, which reflects what is considered good English transliteration in general. Likewise, in English to Bengali (E2B) transliteration, we could find B that maximizes

$$P(B,E) = P(E|B) * P(B) \quad (2)$$

for a given English name. In equations (1) and (2), P(B) and P(E) are usually estimated using n-gram language models. Inspired by research results of grapheme-to-phoneme research in speech synthesis literature, many have suggested phoneme-based approaches to resolving P(B|E) and P(E|B), which approximates the probability distribution by introducing a phonemic representation. In this way, names in the source language, say B, are converted into an intermediate phonemic representation P, and then the phonemic representation is further converted into the target language, say English E. In Bengali to English (B2E) transliteration, the phoneme-based approach can be formulated as P(E|B) = P(E|P) * P(P|B) and conversely we have P(B|E) = P(B|P) * P(P|E) for E2B back-transliteration.

However, phoneme-based approaches are limited by a major constraint that could compromise transliteration precision. The phoneme-based approach requires derivation of proper phonemic representation for names of different origins. One may need to prepare multiple language-dependent grapheme-to-phoneme(G2P) and phoneme-to-grapheme(P2G) conversion systems accordingly, and that is not easy to achieve.

In view of close coupling of the source and target transliteration units, a joint source-channel model, or n-gram transliteration model (TM) has been proposed in (Haizhou *et al.*, 2004). For K alligned transliteration units, we have

$$\begin{aligned} P(B,E) &= P(b_1, b_2.....b_k, e_1, e_2.....e_k) \\ &= P(\langle b,e\rangle_1, \langle b,e\rangle_2, .....\langle b,e\rangle_k) \\ &= \prod_{k=1}^{K} P(\langle b,e\rangle_k \mid \langle b,e\rangle_1^{k-1}) \end{aligned} \quad (3)$$

which provides an alternative to the phoneme-based approach for resolving equations (1) and (2) by eliminating the intermediate phonemic representation.

Suppose that we have a Bengali name $\alpha = x_1x_2.....x_m$ and an English transliteration $\beta = y_1y_2.....y_n$ where $x_i$, i = 1: m are Bengali transliteration units and $y_j$, j = 1: n are English transliteration units. An English transliteration unit may correspond to zero, one or more than one transliteration units in Bengali. Often the values of m and n

are different. There exists an alignment γ with $<b, e>_1 = <x_1, y_1>$; $<b, e>_2 = <x_2x_3, y_2>$; …. and $<b, e>_k = <x_m, y_n>$. A transliteration unit correspondence <b, e> is called a transliteration pair. Thus B2E transliteration can be formulated as

$$\overline{\beta} = \underset{\beta, \gamma}{\operatorname{argmax}} P(\alpha, \beta, \gamma) \quad (4)$$

and similarly the E2B back-transliteration as

$$\overline{\alpha} = \underset{\alpha, \gamma}{\operatorname{argmax}} P(\alpha, \beta, \gamma) \quad (5)$$

An *n*-gram transliteration model is defined as the conditional probability or transliteration probability of a transliteration pair $<b, e>_k$ depending on its immediate *n* predecessor pairs:

$$P(B, E) = P(\alpha, \beta, \gamma) = \prod_{k=1}^{K} P(<b, e>_k \mid <b, e>^{k-1}_{k-n+1}) \quad (6)$$

## Proposed Models

A number of transliteration models have been proposed that can generate the English transliteration from a Bengali word that is not registered in any bilingual or pronunciation dictionary and the vice-versa. The Bengali NE is divided into Transliteration Units (TUs) with patterns $C^+M$, where C represents a consonant or a vowel or a conjunct and M represents the vowel modifier or matra. An English NE is divided into TUs with patterns C*V*, where C represents a consonant and V represents a vowel. The TUs are considered as the lexical units for machine transliteration. The system considers the Bengali and English contextual information in the form of collocated TUs simultaneously to calculate the plausibility of transliteration from each Bengali TU (or English TU) to various English candidate TUs (or Bengali TUs) and chooses the one with maximum probability. This is equivalent to choosing the most appropriate sense of a word in the source language to identify its representation in the target language. The system learns the mappings automatically from the bilingual training sets i.e., corpora guided by linguistic features. The output of this mapping process is a decision-list classifier with collocated TUs in the source language and their equivalent TUs in collocation in the target language along with the probability of each decision obtained from the training sets. The machine transliteration of the input Bengali word is obtained using direct orthographic mapping by identifying the equivalent English TU (or Bengali TU) for each Bengali TU (or English TU) in the input and then placing the English TUs (or Bengali TUs) in order.

Statistical alignment is non-deterministic and requires huge training data as well as smooothing technique. The regular expression based alignment technique has been considerered in the present work as it is deterministic and seems to be more appropriate for English and other Indian languages due to comparable orthography as:

a). English consonants have a corresponding Bengali consonant / conjunct / vowel representation and vice-versa. These are mostly deterministic pairs.

b). English vowels are represented as matra in Indian languages to be attached to characters and vice-versa.

c). The process considers linguistic knowledge in terms of possible conjuncts and diphthongs in Bengali and their corresponding English representations in order to make the TUs on both source and target sides equal. It is completely automatic and no manual intervention is required.

d). No smoothing technique is applied during the TU alignment process. In case of a source TU not present in the alignment, the source language symbols are replaced by the corresponding most probable symbol in the target language. This is considered as the *baseline* of the system.

The various proposed models except the *baseline* model differ in the nature of collocational statistics used during machine transliteration process. All these models (A to F) are basically the variations of the joint source-channel model in respect of the contextual information considered. The models, defined below, are used for Bengali to English machine transliteration. The same models are used during Engish to Bengali machine transliteration also.

● Baseline Model

English consonant or sequence of consonants is represented as Bengali consonant or conjunct or a sequence of consonants. English vowels are represented as either Bengali vowels or as a matra (vowel modifier). English diphthongs are represented as vowel/semi-vowel-matra combination in Bengali.

● Model A

In this model, no context is considered in either the source or the target side. This is essentially the monogram model.

$$P(B,E) = \prod_{k=1}^{K} P(<b,e>_k)$$

● Model B

This is essentially a bigram model with previous source TU, i.e., the source TU occurring to the left of the current TU to be transliterated, as the context.

$$P(B,E) = \prod_{k=1}^{K} P(<b,e>_k \mid b_{k-1})$$

●Model C

This is essentially a bigram model with next source TU, i.e., the source TU occurring to the right of the current TU to be transliterated, as the context.

$$P(B,E) = \prod_{k=1}^{K} P(<b,e>_k \mid b_{k+1})$$

● Model D

This is essentially the joint source-channel model where the previous TUs in both the source and the target sides are considered as the context. The previous TU on the target side refers to the transliterated TU to the immediate left of the current target TU to be transliterated.

$$P(B,E) = \prod_{k=1}^{K} P(\langle b,e\rangle_k \mid \langle b,e\rangle_{k-1})$$

● Model E

This is basically the trigram model where the previous and the next source TUs are considered as the context

$$P(B,E) = \prod_{k=1}^{K} P(\langle b,e\rangle_k \mid b_{k-1}, b_{k+1})$$

● Model F

In this model, the previous and the next TUs in the source and the previous target TU are considered as the context. This is the modified joint source-channel model .

$$P(B,E) = \prod_{k=1}^{K} P(\langle b,e\rangle_k \mid \langle b,e\rangle_{k-1}, b_{k+1})$$

## Bengali to English Machine Transliteration

Translation of named entities is a tricky task: it involves both translation and transliteration. Transliteration is commonly used for named entities, even when the words could be translated: [ওয়াল স্ট্রিট (*wall street*) is translated to *Wall Street* (literal translation) although ওয়াল *(wall)* and স্ট্রিট (*Street)* are vocabulary words]. On the other hand, কল্যানী বিশ্ববিদ্যালয় (*kalyani viswavidyalaya*) is translated to *Kalyani University* in which কল্যানী (*kalyani)* is transliterated to *Kalyani* and বিশ্ববিদ্যালয় (*viswavidyalaya*) is translated to *University.*

Two different bilingual training sets have been kept that contain entries mapping Bengali person names and location names to their respective English transliterations. To automatically analyze the bilingual training sets to acquire knowledge in order to map new Bengali person and location names to English, transliteration units (TUs) are extracted from the Bengali-English pairs of person and location names and Bengali TUs are associated with their English counterparts. Some examples are given below:

[সুনন্দন (*sunandan*) → [সু | ন | ন্দ | ন ], sunandan → [su | na | nda | n ]] , [সিঙ্গুর (*singur*) → [সি | ঙ্গু | র ], singur → [ si | ngu | r ]] , [নন্দীগ্রাম (*nandigramr*) →[ন | ন্দী | গ্রা | ম] , nandigram → [ na | ndi | gra | m ] .

After retrieving the TUs from a Bengali-English pair, it associates the Bengali TUs to the English transliteration units along with the TUs in context. But, in some cases, the number of TUs retrieved from the Bengali and English words may differ. The [বৃজগোপাল (*brijgopal*) ↔ brijgopal ] name pair yields 5 TUs in Bengali side and 4 TUs in English side [ বৃ | জ | গো | পা | ল ↔ bri | jgo | pa | l]. In such cases, the system cannot align the TUs automatically and linguistic knowledge/feature is used to resolve the confusion. The hypothesis followed in the present work is that *the problem TU in the English side has always the maximum length.* If more than one English TU has the same length, then *system starts its analysis from the first one*. In the above example, the TUs *bri* and *jpo* have the same length. The system interacts with the linguistic knowledge and ascertains that *bri* is valid and *jpo* cannot be a valid TU in English since there is no corresponding conjunct representation in Bengali. So *jpo* is split up into 2 TUs *j* and *po*, and the system aligns the 5 TUs as [বৃ | জ | গো | পা | ল ↔ bri | j | go | pa | l]. Similarly, [কোলকাতা (*kolkata*) ↔ kolkata] is initially split as [কো | ল | কা | তা ] ↔ ko | lka | ta], and then as [ ko | l | ka | ta] since *lka* has the maximum length and it does not have any valid conjunct representation in Bengali.

In some cases, the knowledge of Bengali diphthong resolves the problem. In the following example, [সো | মা | লি | য়া *(somalia)* ↔ so | ma | lia], the number of TUs on both sides do not match. The English TU *lia* is chosen for analysis, as its length is the maximum among all the TUs. The vowel sequence *ia* corresponds to a diphthong in Bengali that has the valid representation <ইয়া>. Thus, the English vowel sequence *ia* is separated from the TU *lia* (lia → l | ia) and the intermediate form of the name pair appears to be [সো | মা | লি | য়া ↔ so | ma | l | ia]. Here, a *matra* is associated with the Bengali TU that corresponds to English TU *l* and so there must be a vowel attached with the TU *l.* TU *ia* is further splitted as *i* and *a* (ia → i | a) and the first one (i.e. *i)* is assimilated with the previous TU (i.e. *l*) and finally the name pair appears as: [সো | মা | লি | য়া (*somalia*)↔ so | ma | li | a ]. Similarly, [চে | ন্না | ই (*chennai*) ↔ che | nnai ] and [রা | ই | মা (*raima*) ↔ rai | ma] can be solved with the help of diphthongs.

The number of TUs on both sides doesn't match for the examples, [শি | ব | রা | জ (*shivraj*) ↔ shi | vra | j ], [খ | ড় | দ | হ (*khardah*) ↔ kha | rda | h ]. It is observed that both *vr* and *kd* represent valid conjuncts in Bengali but these examples contain the constituent Bengali consonants in order and not the conjunct representation. During the training phase, if, for some conjuncts, examples with conjunct representation are outnumbered by examples with constituent consonants representation, the conjunct is removed from the linguistic knowledge base and training examples with such conjunct representation are moved to a *Direct example base* which contains the English words and their Bengali transliteration. The above two name pairs can then be realigned as: [শি | ব | রা | জ (*shivraj*) ↔ shi | v | ra | j], [খ | ড় | দ | হ (*khardah*) ↔ kha | r | da | h ].

Otherwise, if such conjuncts are included in the linguistic knowledge base, training examples with constituent consonants representations are to be moved to the *Direct example base*.

The Bengali names and their English transliterations are split into TUs in such a way that, it results in a one-to-one

correspondence after using the linguistic knowledge. But in some cases there exits zero-to-one or many-to-one relationship. Examples of Zero-to-One relationship [Φ → h] are the pairs [আ | ল্লা (*alla*) ↔ a | lla | h] and [মা | ল | দা (*malda*) ↔ ma | l | da | h ], while the pairs [আ | ই | ভি (*aivy*) ↔ i | vy] and [ আ | ই | জ | ল (*aijwal*) ↔ i | zwa | l ] are the examples of Many-to-One relationship [আ, ই→ i]. Also, one-to-zero relationship exists such as: [কৃ | ষ্ণ | ন | গ | র *(krishnanagar)* →kri | shna | ga | r ] [ন → Φ]. These bilingual examples should also be included in the *Direct example base*.

There are some cases where the linguistic knowledge apparently solves the mapping problem, but not always. From the pairs [বরখা (*barkha*) ↔ barkha] and [ঝাড়খন্ড (*jharkhanda*) ↔jharkhand], the system initially generates the mappings [ব | র | খা ↔ ba | rkha] and [রা | জ | খ | ন্ড ↔ jha | rkha | nd] which are not one-to-one. Then it consults the linguistic knowledge base and breaks up the TU as (*rkha → rk | ha*) and generates the final aligned transliteration pairs [ব | র | খা ↔ ba | rk | ha ] and [ঝা | ড় | খ | ন্ড ↔ jha | rk | ha | nd ] (since it finds out that *rk* has a valid conjunct representation in Bengali but not *rkh*), which are incorrect transliteration pairs to train the system. It should have been [ব | র | খা ↔ ba | r | kha] and [ঝা | ড় | খ | ন্ড ↔jha | r | kha | nd ]. Such type of errors can be detected by following the alignment process from the target side during the training phase. Such training examples may be either manually aligned or maintained in the *Direct example base*. Some location names have typical structures and they must be stored in the *Direct example base* in order to get the correct transliterations. The following are the examples of such Bengali-English location pairs:
পলাসী (*palasi*) →palsey, বালিগঞ্জ (*ballyganj*)→ballygunge, বহরমপুর (*baharampur*) → berhampore etc.

The transliteration process described above is concerned with Bengali to English transliteration and this process is applicable to English to Bengali transliteration also.

## Experimental Results

In order to develop the Bengali-English machine transliteration system, two different databases containing 7200 person names and 5100 location names have been collected and their corresponding English transliterations have been stored manually. The proposed models of the transliteration system generate the collocational statistics from these two databases. These statistics serve as the decision list classifier to identify the target language TU given the source language TU and its context.

Each database is initially distributed into 10 subsets of equal size. In the open test, one subset is withheld for testing while the remaining 9 subsets are used as the training sets. This process is repeated 10 times to yield an average result, which is called the 10-fold open test. After the experiments, it has been found that each of the 10-fold open tests gave consistent error rates with less than 1% deviation. Therefore, for simplicity, we have randomly selected one of the 10 subsets as the standard open test to report the results. The test sets of person and location names consist of 720 and 510 entries respectively. All the proposed models along with the *baseline model* have been evaluated with these test sets. Some statistics about the two different test sets are presented in Table 1.

The models are evaluated on the basis of the two evaluation metrics, Word Agreement Ratio (WAR) and Transliteration Unit Agreement Ratio (TUAR). The evaluation parameter Character Agreement Ratio in (Goto *et al*. 2003) has been modified to Transliteration Unit Agreement Ratio as vowel modifier matra symbols in Bengali words are not independent and must always follow a consonant or a conjunct in a Transliteration Unit. Let, B be the input Bengali word, E be the English transliteration given by the user in open test and $E^{/}$ be the system-generated transliteration. TUAR is defined as, TUAR = (L-Err)/ L, where L is the number of TUs in E, and Err is the number of wrongly transliterated TUs in $E^{/}$ generated by the system. WAR is defined as, WAR= (S-$Err^{/}$) / S, where S is the test sample size and $Err^{/}$ is the number of erroneous names generated by the system (when $E^{/}$ does not match with E).

The results of the tests in terms of evaluation metrics have been presented in Table 2 for person and location names for Bengali to English (B2E) transliteration. The modified joint source-channel model (Model F) exhibits the best performance with a Word Agreement Ratio (WAR) of 74.4% and a Transliteration Unit Agreement Ratio (TUAR) of 91.7% for person names. The same model (Model F) also demonstrated the highest WAR of 72.6% and TUAR of 89.3% for location names. The joint source-channel model (Model D) has not performed well in the Bengali-English (B2E) machine transliteration for person and location names whereas the trigram model (Model E) needs further attention as its result are comparable to the modified joint source-channel model (Model F).

All the models have also been tested for back-transliteration, i.e., English to Bengali transliteration (E2B). The results of the E2B transliteration system in terms of the evaluation metrics WAR and TUAR have been shown in Table 3 for person and location names. It is observed that the modified joint source-channel model (Model F) performs best in back-transliteration with a WAR of 72.5% for person names, 70.5% for location names and a TUAR of 90.8% for person names and 87.6% for location names. Here, in this case, joint source channel model (Model D) did not yield better results whereas the results of the trigram model (Model E) are impressive compared to the modified joint source channel model (Model F).

Table 1: Statistics of the test sets

| Test set type | Sample size (S) | No. of TUs (L) | Average no. of TUs per name |
|---|---|---|---|
| Person name | 720 | 3615 | 5 |
| Location name | 510 | 2018 | 4 |

Table 2: Results with Evaluation Metrics for Person and Location names (B2E transliteration)

| Model | Person Name | | Location name | |
|---|---|---|---|---|
| | WAR (in %) | TUAR (in %) | WAR (in %) | TUAR (in %) |
| Baseline | 49.7 | 74.8 | 47.1 | 73.9 |
| A | 53.8 | 79.2 | 53.3 | 77.1 |
| B | 63.4 | 83.3 | 62.8 | 81.2 |
| C | 60.7 | 82.5 | 60.1 | 80.7 |
| D | 65.8 | 84.9 | 64.3 | 82.2 |
| E | 70.6 | 89.3 | 68.9 | 86.9 |
| F | 74.4 | 91.7 | 72.6 | 89.3 |

Table 3: Results with Evaluation Metrics for Person and Location names (E2B transliteration)

| Model | Person name | | Location name | |
|---|---|---|---|---|
| | WAR (in %) | TUAR (in %) | WAR (in %) | TUAR (in %) |
| Baseline | 49.7 | 74.8 | 47.1 | 73.9 |
| A | 53.4 | 77.2 | 51.6 | 79.8 |
| B | 60.4 | 81.7 | 58.2 | 83.8 |
| C | 57.5 | 79.2 | 54.9 | 82.2 |
| D | 61.7 | 83.3 | 59.5 | 84.3- |
| E | 67.2 | 87.5 | 66.3 | 85.2 |
| F | 72.5 | 90.8 | 70.5 | 87.6 |

## Conclusion

The modified joint source-channel model is general and can be applied for any languages. The TU alignment process is applicable for languages that share a comparable orthography with English like Bengali and other Indian languages. Because of this comparable orthography, it is possible to obtain reasonable results with a small sample. It has been observed that the modified joint source-channel model performs best in terms of WAR and TUAR. Detailed examination of the evaluation results reveals that Bengali has separate short and long vowels and the corresponding matra representation while these may be represented in English by the same vowel. It has been observed that most of the errors are at the *matra* level i.e., a short matra might have been replaced by a long matra or vice versa. More linguistic knowledge is necessary to disambiguate the short and the long vowels and the matra representations in Bengali. Besides person and location names, organization names are also to be used for training and testing the proposed models**.**

## References


Al-Onaizan, Y., and Knight, K. 2002a. Named Entity Translation: Extended Abstract. In *Proceedings of the Human Language Technology Conference.* 122-124.

Al-Onaizan, Y., and Knight, K. 2002b. Translating Named Entities Using Monolingual and Bilingual Resources. In *Proceedings of the 40th Annual Meeting of the ACL*. 400-408.

Al-Onaizan, Y., and Knight, K. 2002c. Machine Transliteration of Names in Arabic Text. In *Proceedings of the ACL Workshop on Computational Approaches to Semantic Languages*.

Arbabi Mansur; Scott M. Fischthal; Vincent C. Cheng; and Elizabeth Bar. 1994. Algorithms for Arabic name transliteration. *IBM Journal of Research and Development* 38(2):183-193

Crego, J. M.; Marino, J.B.; and Gispert, A. de. 2005. Reordered Search and Tuple Unfolding for Ngram-based SMT. In *Proceedings of the MT-Summit X*. 283-289.

Goto, I.; Kato, N.; Uratani, N.; and Ehara, T. 2003. Transliteration considering Context Information based on the Maximum Entropy Method. In *Proceeding of the MT-Summit IX.*125–132.

Haizhou Li; Zhang Min; and Jian, Su. 2004. A Joint Source-Channel Model for Machine Transliteration. In *Proceedings of the ACL 2004*. 159-166.

Knight, K., and Graehl, J. 1998. Machine Transliteration, Computational Linguistics 24(4): 599-612.

Marino, J. B.; Banchs, R.; Crego, J. M.; Gispert, A. de; Lambert, P.; Fonollosa, J. A.; and Ruiz, M. 2005. Bilingual N-gram Statistical Machine Translation. In *Proceedings of the MT-Summit X.* 275-282.

Meng Helen M.; Wai-Kit Lo; Chen, Berlin; and Tang, Karen. 2001. Generating Phonetic Cognates to handle Name Entities in English-Chinese Cross-language Spoken Document Retrieval. In *Proceedings of the Automatic Speech Recognition and Understanding (ASRU) Workshop.*

Nasreen, A. J., and Larkey, Leah S. 2003. Statistical Transliteration for English-Arabic Cross Language Information Retrieval. In *Proceedings of CIKM*. 139-146.

Paola, Virga, and Khudanpur, Sanjeev. 2003. Transliteration of Proper Names in Crosslingual Information Retrieval. In *Proceedings of the ACL 2003 Workshop on Multilingual and Mixed-language Named Entity Recognition*. 57-60.

Stalls, Bonnie Glover and Kevin Knight. 1998. Translating names and technical terms in Arabic text. In *Proceedings of COLING/ACL Workshop on Computational Approaches to Semantic Languages*, Montreal, Canada. 34-41.